COMPARISON OF OPTIMIZATION TECHNIQUES FOR COMPLEX SUPPLY CHAIN NETWORK PLANNING PROBLEMS

Luis Filipe de A. Pessoa, Dennis Horstkemper, Diego de Siqueira Braga, Bernd Hellingrath

Westfälische Wilhelms-Universität Münster
Chair for Information Systems and Supply Chain Management

Marcelo Gomes Pereira de Lacerda, Fernando Buarque de Lima Neto

Universidade de Pernambuco
Escola Politécnica de Pernambuco

RESUMO

Atividades de planejamento no Gerenciamento de Cadeias de Suprimento são complexas devido a suas diversas influências, como os efeitos da globalização, customização em massa de produtos e curto ciclo de vida dos produtos. Isso conduz a requerimentos adicionais para o planejamento logístico em geral. O planejamento e gerenciamento de cadeias de suprimento é um recurso fundamental para as empresas atuais, a fim de aumentar a rentabilidade e diminuir a incerteza e o risco de determinadas ações tomadas por um ou mais membros da cadeia de suprimento. Nesse contexto, o Planejamento de Redes de Cadeias de Suprimento (SCNP) é usado para determinar uma alocação adequada de demandas a um custo mínimo. Se aplicado no modelo de planejamento no nível tático, SCNP pode ser usado como uma importante entrada para tarefas de planejamento operacional, como é o caso do planejamento mais detalhado de transporte. Devido à complexidade do problema, técnicas matemáticas para obtenção de soluções exatas podem consumir uma grande quantidade de tempo e, por isso, não podem ser aplicadas na prática. Assim, sistemas de suporte de decisão em logística, como Sistemas de Planejamento Avançado (APS) incorporam heurísticas ou oferecem abordagens de soluções metaheurísticas, a fim de proporcionar boas soluções em prazos mais curtos. Entretanto, há uma necessidade constante de técnicas novas e mais eficientes para tratar melhor a crescente complexidade de cadeias de suprimento modernas e para lidar com ciclos de planejamento cada vez mais curtos. Portanto, este trabalho procura explorar as possibilidade de aplicar técnicas de otimização metaheurísticas em problemas SCNP.

ABSTRACT

Planning tasks in Supply Chain Management demonstrate a high complexity due to several influences such as globalization effects, mass customization of products and shorter product life cycles. This is leading to additional requirements for logistics planning in general. Nevertheless, the planning and management of supply chains is a key capability for today’s companies in order to increase profitability as well as decrease the uncertainty and the risk of certain actions taken by one or more members of the supply chain. In this context, Supply Chain Network Planning (SCNP) is used to determine an adequate allocation of demands and capacities in storage, transportation and production facilities, aiming to fulfill customer demands at minimal costs. As a planning model on the tactical level, SCNP can be used as an important input for operational planning tasks, e.g. for a more detailed transport planning. However, due to the complexity of the problem, exact mathematical solution techniques can consume a high amount of time, and are as such not applicable in practice. Logistics decision support systems, like Advanced Planning Systems (APS) extend them therefore with heuristics or offer meta-heuristic solution approaches, each in order to provide good solutions in short timeframes. Despite this, there is a constant need for new, more efficient techniques to tackle the increasing complexity of modern supply chains and to handle ever shorter planning cycles. Therefore, this paper seeks to explore the possibilities of applying modern meta-heuristic optimization techniques onto SCNP problems.

1. INTRODUCTION

The application of analytic and quantitative methods in logistics and Supply Chain Management is a research area in which new advances are constantly made. However, modern supply chains show an ever increasing complexity due to several influences such as heterogeneity in customer needs, shorter product lifecycles and a globalization of the supplier base (Bozarth et al., 2009). Thus, the management of such complex inter-organizational networks is a challenging key capability for today's companies.

In this context, the Supply Chain Network Planning (SCNP) determines an adequate allocation of demands and capacities in storage, transportation and production facilities, aiming to fulfill customer demands at minimal costs (Simchi-Levi et al., 2007). As a planning task on the tactical level, the resulting plan is an important input to operational planning tasks such as transportation planning and can be used to make sure, that sufficient capacities are available at the right time. The medium–term planning horizon allows for some changes in capacities during the operation of the supply chain, e.g. by the introduction of overtimes or the contracting of additional logistics service providers, and therefore the SCNP can be used to determine plans which avoid capacity bottlenecks. The different facilities in the supply chain are interdependent: The appropriate determination of transportation capacities also necessitates a corresponding determination of production capacities. Therefore, all the aforementioned influences, which increase the complexity of supply chains as a whole, also increase the complexity of SCNP problems. If bottlenecks occur during transportation in particular, the following decisions to deal with this issue can be performed by the SCNP, according to (Rohde and Wagner 2005): (1) Produce and ship in earlier periods while increasing seasonal stock in a distribution center, (2) distribute products using alternative transportation modes with different capacities and costs and (3) deliver to customers from another distribution center. However, in many implementations of this planning problem not all of these decisions are taken into account with all possible variations, e.g. only one possible transportation mode might be considered when linking two nodes in the supply chain together and alternative modes are only considered, when a different transport route is taken as well. The inclusion of all these options for the tactical planning of meaningful transportation decisions leads to an increased complexity as additional integer variables have to be introduced into the decision model to represent the discrete transport amounts and the different transportation modes, leading to an increased complexity. Therefore a compromise between the included planning decisions and the capabilities of the solution techniques has to be found (Stadtler, 2005). Considering these requirements, efficient solution techniques, which are capable to handle this additional requirements and the constant growth in planning complexity, are necessary. Thus, this paper seeks to explore the applicability of several modern solution techniques, which can quickly find good and applicable solutions, onto complex SCNP problems, while using a well-established formulation of this problem. If this problem can be solved by these new methods in an efficient way, it will be possible to include additional planning decisions, e.g. the ones depicted above, into the problem formulation and therefore enhance its' value for decision makers in logistics and the supply chain as a whole. One instance of such modern solution methods are meta-heuristics, which employ problem independent approaches and can therefore be applied on a multitude of different problems. Most of them scale well with the size of the problems and require only little work in their parameterization. In this regard meta-heuristics are then more flexible than problem specific approaches, i.e. customized heuristics.

This paper is structured as follows: firstly, a mathematical model of a Supply Chain Network Planning problem is presented as a model from the logistics domain. Afterwards, the five different solution techniques, which are applied to the SCNP within this work, are shortly depicted. Within chapter IV the implementation approach for the application of the five different meta-heuristics for the SCNP is shown. Chapter V presents the results of the application upon a complex test scenario. Lastly, the results of the paper are summarized and an outlook towards further research options is given.

2. MATHEMATICAL MODEL OF THE PLANNING PROBLEM

A mathematical model of the investigated decision problem, the SCNP, is presented in this section. The model was based on the work by Hellingrath and Küppers (2011), which in turn is an adaptation from the works of Pibernik and Sucky (2005) which concerned the same domain. As described in chapter 1, this model does not include all possible decision variables, e.g. the choice of transport modes on a specific route. Table 1 describes the used variables and parameters.

The objective function (1) of the model describes a minimization of all costs. It includes production costs, setup costs, costs for capacity adjustment measures in each facility as well as transportation and inventory holding costs. Furthermore, the model contains a variable called “demand fulfillment”. It is used to determine the relative volume of fulfilled and therefore also unfulfilled end-customer demands. The relative amount of unfulfilled demands is multiplied with fictional backorder costs as a penalty. The restrictions (2) and (3) ensure the correct flow of materials throughout the network. Restriction (4) represents the need to fulfill final customer demand by the last stage of the node and is used to determine the “demand fulfillment” factor. Restrictions (5), (6), and (7) provide that maximum capacities for the production, storage and transportation of goods have to be met. To simplify the model description, further restrictions for the non-negativity of variables and the correct setting of helping variables have been omitted.

$$min \sum_{t=1}^{T} \sum_{i=1}^{I} \sum_{k \epsilon K_i} \sum_{n \epsilon N_i} \left[PQ_{i,k,t}^{n} \cdot ProdCst_{i,k}^{n} + z_{i,k,t}^{n} \cdot FixCst_{i,k}^{n} + \sum_{f \epsilon F_{i,k}} y_{i,k,t}^{f} \cdot PCst_{i,k}^{f} + \sum_{l \epsilon K_{i+1}} TQ_{i,k,l,t}^{n} \cdot TransCst_{i,k,l}^{n} + IQ_{i,k,t}^{n} \cdot InvCst_{i,k}^{n} + \sum_{m \epsilon N_{i-1}} IQ_{i,k,t}^{m} \cdot InvCst_{i,k}^{m} \right] + \sum_{t=1}^{T} \sum_{n \epsilon N_I} (1 - df_{n,t}) \cdot BOCst_n \tag{1}$$

Subject to:

$$IQ_{i,k,t}^{n} = IQ_{i,k,t-1}^{n} + PQ_{i,k,t}^{n} - \sum_{l \epsilon K_{i+1}} TQ_{i,k,l,t}^{n} \qquad \forall\, i, n \,\epsilon\, N_i, k \,\epsilon\, K_i, t \tag{2}$$

$$IQ_{i,k,t}^{m} = IQ_{i,k,t-1}^{m} - \sum_{n \epsilon N_i} PQ_{i,k,t}^{n} \cdot BOMF_{i,m,n} + \sum_{l \epsilon K_{i-1}} TQ_{i-1,l,k,t}^{m} \qquad \forall\, i, m \,\epsilon\, N_{i-1}, k \,\epsilon\, K_i, t \tag{3}$$

$$\sum_{k \epsilon K_I} \sum_{l \epsilon K_{i+1}} TQ_{i,k,l,t}^{n} = df_{n,t} \cdot DM_{I,t}^{n} \qquad \forall\, n \,\epsilon\, N_I, l \,\epsilon\, K_{I+1}, t \tag{4}$$

$$\sum_{n \epsilon N_i} PQ_{i,k,t}^{n} \cdot pc_k^n \le BC_{i,k,t}^{prod} + \sum_{f \epsilon F_{i,k}} y_{i,k,t}^{f} \cdot incC_{i,k}^{f} \qquad \forall\, i, k, t \tag{5}$$

$$\sum_{n \epsilon N_i} IQ_{i,k,t}^{n} \cdot ic_k^n + \sum_{m \epsilon N_{i-1}} IQ_{i,k,t}^{m} \cdot ic_k^m \le BC_{i,k,t}^{inv} \qquad \forall\, i, k, t \tag{6}$$

$$\sum_{n \epsilon N_i} TQ_{i,k,l,t}^{n} . tc_{k,l}^{n} \le BC_{i,k,l,t}^{trans} \qquad \forall\, i, k \,\epsilon\, K_i, l \,\epsilon\, K_{i+1}, t \tag{7}$$

Table 1: Description of Variables

Var.	Description	Var.	Description
$\boldsymbol{t}$	Period	$\boldsymbol{I}$	SC tier (i=1,…,I+1)
$\boldsymbol{InvCst_{i,k}^{m}}$	Inventory holding costs for intermediate product m on node k (tier i)	$\boldsymbol{df_{n,t}}$	Fulfillment of the final customer demand of final product n in period t (in percent)
$\boldsymbol{K_i}$	Set of nodes on tier i	$\boldsymbol{M}$	Large number
$\boldsymbol{Ni}$	Set of products produced on tier i	$\boldsymbol{IQ_{i,k,t}^{n}}$	Inventory quantity of final product n on node k (tier i) in period t
$\boldsymbol{PQ_{i,k,t}^{n}}$	Production quantity of product n on node k (tier i) in period t	$\boldsymbol{BOMF_{i,m,n}}$	Required quantity of intermediate product m (from tier i-1) for the production of final product n (bill-of-material factor)
$\boldsymbol{ProdCst_{i,k}^{n}}$	Production costs of product n (tier i) and node k	$\boldsymbol{DM_{I,t}^{n}}$	Ultimate customer demand for product n in period t
$\boldsymbol{z_{i,k,t}^{n}}$	Binary variable indicating that product n is produced on node k (tier i) in period t	$\boldsymbol{pc_{i,k}^{n}}$	Required production capacity for the production of product n on node k (tier i)
$\boldsymbol{FixCst_{i,k}^{n}}$	Changeover costs for the production of product n and node k (tier i)	$\boldsymbol{BC_{i,k,t}^{prod}}$	Production capacity of node k (tier i) in period t (basic capacity)
$\boldsymbol{F_{i,k}}$	Set of adjustment measures available on node k (tier i)	$\boldsymbol{IQ_{i,k,t}^{m}}$	Inventory quantity of intermediate product m on node k (tier i) in period t
$\boldsymbol{y_{i,k,t}^{f}}$	Binary variable indicating that adjustment measure f is implemented on node k (tier i) in period t	$\boldsymbol{incC_{i,k}^{f}}$	Production capacity increase adjustment measure f on node k (tier i)
$\boldsymbol{TQ_{i,k,lt}^{n}}$	Transport quantity of product n from node k (tier i) to node l (tier i+1)	$\boldsymbol{ic_{i,k}^{n}}$	Required inventory capacity of product n on node k (tier i)
$\boldsymbol{TransCst_{i,k,l}^{n}}$	Transport costs for product n from node k on (tier i) to node l (tier i+1)	$\boldsymbol{ic_{i,k}^{m}}$	Required inventory capacity of intermediate product m on node k (tier i)
$\boldsymbol{BOCst_n}$	Costs for unfulfilled demand of product n	$\boldsymbol{BC_{i,k,t}^{inv}}$	Inventory capacity of node k (tier i) in period t
$\boldsymbol{BC_{i,k,l,t}^{trans}}$	Transport capacity between node k (tier i) and node l (tier i+1) in period t		

3. META-HEURISTICS AS AN OPTIMIZATION TECHNIQUE FOR SCNP

Metaheuristics have been applied in many logistics areas to solve complex problems within all planning levels, i.e. operational, tactical and strategic. However, a review shows, that between 2000 and 2012, only 15% of the papers in analytic work within the *Journal of Business Logistics* applied heuristics and metaheuristics, while 33% applied exact optimization techniques and 59% applied simulation methods (Griffis et al., 2012). Although the number of approaches which applied metaheuristics to solve logistic problems is less representative than those of the other methods, this is a promising area, given that in recent decades the number of approaches which use this techniques is increasing. Based on further results presented by Griffis et al. (2012), it is possible to see that the number of approaches which made use of metaheuristics to solve complex problems in logistics is increasing along the years: 33 papers were published between 1990 and 1999; 67 between 2000 and 2009; and, 28 between 2010 and 2012. This means an increase of 100% in the decade of 2000, and an increasing of 75% considering the first three years in the 2010 decade compared to the last one. Those approaches can also be classified within the logistics levels as follows: 84 (67%) from operational level; 22 (18%) from tactical level; and 28 (23%) from strategic level - the percentage sum is greater than one hundred because some approaches belong to more than one level. This motivates the extension of the investigated research especially in the less investigated area of tactical planning.

Considering the papers presented within the literature research conducted by Griffis et al. (2012), most of the approaches have used traditional metaheuristic techniques in logistics problems, such as Tabu Search, Genetic Algorithm and Simulated Annealing. Furthermore, the research in metaheuristic applications on logistics planning problems is especially vivid in current times, for example Ojha et al. (2013) tackled a fuzzy Multi-Item Balanced Transportation Problem (MIBTP) subjected to imprecise transportation costs, warehouse and budget constraints at destination; Brunwal and Deb (2013) developed an approach to solve a scheduling optimization problem of a Flexible Manufacture System (FMS) and Marinakis et al. (2013) solved a Vehicle Routing Problems with Stochastic Demands (VRPSD). Some of these papers also apply more advanced metaheuristic techniques than the traditionally established ones.

One of the reasons for the increasing number of solution approaches that use metaheuristic is that exact mathematical methods, like Mixed Integer Programming (MIP), can entail runtimes, which are inapplicable in practice, when lots of decision variables are incorporated into the planning task. Moreover, metaheuristics enable decision support systems in logistics to find solutions for complex problems by an intelligent way, since they might incorporate a self-adaptation of their parameters, a short, medium and long term memory, machine learning tools and expert knowledge to improve their performance (Melián-Batista and Moreno-Perez, 2013). According to Melián-Batista and Moreno-Perez (2013), although there is no guarantee that the optimum value will be achieved, metaheuristics are among the most efficient and robust methods to find high-quality solutions (i.e. close from the optimal solution) for several complex optimization problems and. For this reason, intelligent approaches to deal with real-world problems in logistics planning often include metaheuristics.

Ponsignon and Mönch (2012) proved that the SCNP is a NP-hard problem, which is a sufficient condition for the application of metaheuristics. Therefore, they also created a metaheuristic solution approach for a specialized case of the SCNP in the semiconductor industry based on a GA. Furthermore, Soares and Vieira (2009) created a multiobjective implementation for the SCNP using GA as well. These works show that this line of research should be investigated further, which is the aim of this paper.

The constant development in the area of metaheuristics led to the emergence of a multitude of new approaches (e.g. Fish School Search (FSS) (Bastos-Filho et al., 2009)) which can be applied for a large set of complex problems. Moreover, many improvements of established metaheuristics (like the particle Swarm optimization (PSO) (Kenney and Eberhart, 1995) and Differential Evolution (DE) (Storn and Price, 1997)) have been proposed which enable those techniques to self-adapt their parameters (e.g. Self-Adaptive Differential Evolution (SaDE) (Qin and Suganthan, 2005)), or to better address the exploitation-exploration trade-off problem (e.g. Heterogeneous PSO (HPSO) (Engelbrecht, 2010)). All these mentioned improvements aim at solving more complex problems and these algorithms need to be evaluated to assess their applicability for logistics planning problems and their rise in requirements, which is why this paper aims to assess their applicability on the SCNP.

The PSO algorithm is a population based meta-heuristic inspired by the collective behavior of birds, which are represented using particles. In PSO, each particle aims at finding solutions to a given problem through a movement mechanism that uses collective and local information. This approach is well known as an approach with a fast convergence towards a possible solution. However, it is likely to get stuck in local minima in complex search spaces (Kennedy and Eberhart, 1995). Differential Evolution (DE) is a population based search algorithm, a technique of Evolutionary Computing, this algorithm aims at the optimization of nonlinear and non-differentiable functions in a continuous space. A great advantage of DE is the use of small amount of parameters (i.e. population size, differential weight and crossover probability).

An important aspect of search algorithms is the ability of balancing between exploration and exploitation within a given search space, i.e. the set of all possible solutions for a given problem. The common idea is to make the algorithm perform an exploration in the beginning of the search process and slowly change the collective behavior to exploitation, performing a wider search process at the start iterations and to converge to a certain area within the search space in the end. However, an interchanging process between both behaviors can improve the performance of the algorithms, but it is hard to determine whether this transition should be done. Aiming to create an adaptive behavior between exploration and exploitation, Engelbrecht (2010) proposed and adaptive heterogeneous version of the PSO, called HPSO, where particles can perform different behaviors during the search process. The change of mechanisms is determined by the history of the search process of each particle. This way, particles are able to move from exploration to exploitation and visa-versa whenever needed.

For complex problems, it is hard to define values for the parameters of any meta-heuristics. Moreover, it is even harder to define these parameters in problems which are positioned within a search space, of which the decision maker only has limited knowledge. This is the case for the SCNP, as its search space vastly varies with the exact parameterization based on a specific planning situation. The adequate values for these parameters are very often strongly dependent on the problem and its instance. Therefore, in an attempt to overcome this problem, SaDE dynamically adjusts two out of three critical parameters of DE during the search process. Implicitly, this adaptation not only gives the algorithm the ability to adjust itself to the given problem, but also enables it to better balance between exploration and exploitation.

Bastos-Filho et al. (2009) proposed a novel meta-heuristic inspired in the collective foraging behavior of natural fish schools. This meta-heuristic is also inspired by the PSO. In the FSS, the success of the search process is represented by the weight of the fishes, which means that a good search history of this fish also increases the weight of a fish. A second means to encode success in FSS is the radius of the school. The school expands throughout the search space when no good solutions can be found, in order to look for better ones in other places. When some fishes increase their weight, the fish school contracts itself towards their position and the different individuals will examine the area close to these heavier fishes in more detail, due to the high likelihood of having good solutions in this area. These operations, albeit simple in nature, are different from the PSO and describe another effective way to deal with the exploration against exploitation tradeoff.

4. COMPUTATIONAL EXPERIMENTS

The metaheuristics described in Section 3 were applied on a SCNP problem derived from a real world case, as shown in figure 1, aiming at (i) testing the applicability of these approaches for SCNP and (ii) to assess the performance of each approach in consideration of the logistical viability. To ease the implementation, product groups instead of single products were considered. The proposed scenario describes the supply chain of a convenience product manufacturer and is composed of three production facilities (a paper-mill, a supplier and the manufacturers production plant), one international distribution center (IDC) and four regional distribution centers (RDC), as shown in Figure 1. The plant produces ten different product groups, which are based on two different kinds of materials, which in turn are supplied by a paper mill, and delivers these manufactured products to the IDC. The IDC also receives another eight product groups from an external supplier and distributes the products of both the production plant and the supplier to the different RDCs. The planning horizon has been set to 6 months with dynamic end-customer demands.

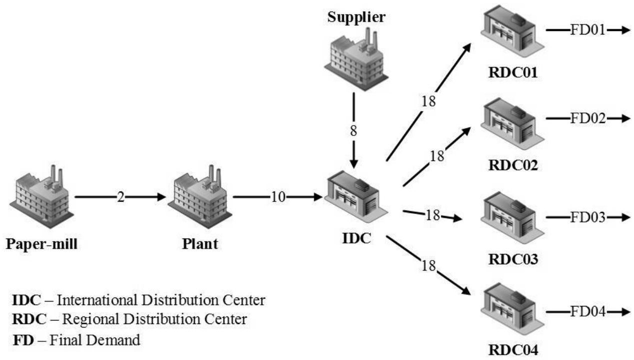

Figure 1: Testing Scenario

To evaluate the applicability of the different metaheuristics, a prototypical implementation has been created. It adheres to all constraints expressed in the mathematical model presented in chapter 2 and also considers the same cost types. In accordance to related work performed on the SCNP (Ponsignon & Mönch, 2012) and similiar problems, only a subset of the variables in the SCNP is generated by the FSS algorithm itself: All production and transportation volumes are determined by the FSS, while all additional variables are calculated based on the models constraints and equations, i.e. inventory levels for final products in a production facility are determined by adding production volumes to the inventory levels of the previous planning period and subtracting the outgoing transportation volumes. Initial values for all variables in the solution vectors are generated randomly. The solution vector for this problem, concerning the scenario shown in section 4, which is computed by the metaheuristics, contains 672 variables, which encompass all product and transportation quantities for all periods. All decision variables not represented in the solution vector of the meta-heuristic are heuristically calculated, i.e. inventory volumes for final products in a certain facility can be calculated by adding production volumes and incoming transportation volumes to the inventory volumes of the same product in the same facility of the previous period while also subtracting demands and outgoing transport volumes. Given that the selected metaheuristics do not incorporate a mechanism to deal with constraints, penalty cost factors were incorporated into the objective function in order to avoid infeasible solutions (e.g. regarding transportation capacity). In general, a constraint in the form of $a < b$ can be reformulated into $a+c = b$. Here, c measures the *degree of infeasibility*. Including the product of this new variable and an appropriately sized penalty cost factor into the objective function of the problem forces the metaheuristics to avoid infeasible solutions.

Based on several experiments, the best parameter configuration and, in case of the HPSO also the particle behavior, was selected for each metaheuristic, in order to find an efficient configuration for each approach. All algorithms were run 30 times with 10 and 100 individuals and 100, 1000, 10000 and 100000 iterations. The results for each experiment will be compared in the next section. It is important to highlight that, as the FSS makes two fitness function calls per iteration (Bastos-Filho et al., 2009), the number of iterations used for the FSS in each experiment will be the half of the values for the other techniques.

5. RESULTS

Table 2 shows the results for all algorithms in all experiments. The column *Individuals* indicates the number of individuals, which were used in each experiment; the column *Iterations* shows the number of performed iterations; the column *Best Fitness* shows the best result found by the 30 executions of each experiment; the columns *Mean* and *Standard deviation* shows the statistical values for the best result achieved by each experiment execution. The best results for each metric are in bold. Regarding the best fitness, which is the best solution found among the 30 simulations, FSS and SaDE obtained the best result in 3 out of 9 experiments, being the most successful algorithms. FSS and DE reached the best results in two experiments each, while the HPSO overcame the other techniques in only one case and the PSO in none of them. For the mean cost, SaDE overcame the other algorithms in 4 out of 9 experiments. DE beat the other ones in two of them, while FSS showed the best performance in three cases. The PSO and HPSO algorithms did not have the best results in any of the cases. Finally, the FSS showed the best performance for standard deviation in 6 out of 9 cases, while DE and HPSO showed the most constant results in only one experiment.

Table 2: Experiments' results for all algorithms.

Individuals	Iterations	PSO		
		Best Fitness	**Mean**	**Standard Deviation**
10	100	1.46E+09	1.56E+09	5.89E+07
10	1000	1.55E+09	1.65E+09	5.91E+07
10	10000	1.54E+09	1.59E+09	4.74E+07
10	100000	1.49E+09	1.61E+09	6.46E+07
100	100	9.42E+08	1.07E+09	9.40E+07
100	1000	1.03E+09	1.08E+09	3.40E+07
100	10000	1.05E+09	1.13E+09	5.13E+07
100	100000	9.36E+08	9.86E+08	4.81E+07
100	1000000	9.27E+08	9.93E+08	1.61E+07
Individuals	**Iterations**	**HPSO**		
		Best Fitness	**Mean**	**Standard Deviation**
10	100	1.68E+09	2.10E+09	1.47E+08
10	1000	**1.40E+08**	1.50E+09	6.32E+08
10	10000	2.94E+07	2.98E+07	**2.37E+05**
10	100000	2.42E+07	2.67E+07	1.49E+06
100	100	1.97E+09	2.02E+09	2.93E+07
100	1000	7.02E+08	1.19E+09	3.53E+08
100	10000	2.10E+07	2.48E+07	2.73E+06
100	100000	8.64E+06	9.14E+06	5.04E+05
100	1000000	7.69+06	8.02E+06	3.66E+05

Continued.

(Continued)

Individuals	Iterations	DE		
		Best Fitness	**Mean**	**Standard Deviation**
10	100	1.98E+09	2.15E+09	8.07E+07
10	1000	9.40E+08	1.01E+09	4.36E+07
10	10000	**1.24E+07**	**1.40E+07**	1.07E+06
10	100000	7.41E+06	9.54E+06	7.67E+05
100	100	1.96E+09	2.14E+09	7.73E+07
100	1000	8.15E+08	9.85E+08	6.15E+07
100	10000	**9.37E+06**	**1.26E+07**	**1.34E+06**
100	100000	5.58E+06	7.94E+06	9.17E+05
100	1000000	5.46E+06	7.47E+06	5.51E+05
Individuals	**Iterations**	**SaDE**		
		Best Fitness	**Mean**	**Standard Deviation**
10	100	**9.89E+08**	**1.37E+09**	2.25E+08
10	1000	2.96E+08	**3.36E+08**	3.66E+07
10	10000	2.38E+07	7.22E+07	7.88E+07
10	100000	2.35E+07	4.50E+07	2.94E+07
100	100	**6.67E+08**	**7.27E+08**	4.33E+07
100	1000	**1.39E+08**	**1.49E+08**	7.07E+007
100	10000	1.91E+07	2.04E+07	1.36E+06
100	100000	1.66E+07	1.76E+07	9.95E+05
100	1000000	9.64E+06	1.47E+07	8.84E+05
Individuals	**Iterations**	**FSS**		
		Best Fitness	**Mean**	**Standard Deviation**
10	50	1.44E+09	1.51E+09	**4.25E+07**
10	500	5.61E+08	6.19E+08	**3.30E+07**
10	5000	1.10E+08	1.34E+08	8.25E+06
10	50000	**6.51E+06**	**8.49E+06**	**7.59E+05**
100	50	9.84E+08	1.05E+09	**2.87E+07**
100	500	1.88E+08	2.19E+09	**1.48E+07**
100	5000	1.13E+07	1.44E+07	2.25E+06
100	50000	**1.83E+06**	**1.98E+06**	**9.77E+04**
100	500000	**9.45E+06**	**9.56E+06**	**1.03E+05**

The aforementioned results showed that SaDE is able to find good master plans in most of the cases in which there are not many available numbers of iterations. However, much better solutions are achieved by FSS and DE runs with more iterations. It is important to notice that, from 5000 to 50000 iterations, the FSS’s performance is strongly improved, showing that this algorithm needs more available time to beat DE. This is due to the characteristic of the FSS to slowly converge but to find a better solution than the other algorithms, if sufficient time is available. It is also important to highlight that the adaptive algorithms had clearly better results than their standard versions. One possible cause is the ability of these techniques to balance the search process between exploration and exploitation during the optimization.

Figure 2 shows the cost values of the master plans generated by the metaheuristics in relation to the required number of iterations. In a short timeframe, the approaches based on Evolutionary Computing (DE and SaDE) find a better configuration for the master plan than the others algorithms, but they cannot minimize the costs during the further iterations as effective as during the initial runtime. This problem, known as anticipated convergence, is even worse for the PSO, which got stuck into local minima early within the optimization procedure. Although the FSS does not optimize the master plan within the initial iterations as fast as DE and SaDE, it is able to find a SC configuration which considerably minimizes the costs even before 10% of the planned runtime (i.e. the number of iterations). Moreover, FSS was the only technique that managed to constantly minimize the cost until the end of the optimization process. This shows that this metaheuristic is able to create the most cost efficient and applicable plans for the SCN out of this selection, considering all the requirements mentioned for this problem, i.e. (i) to find the best master plan which minimizes the SCN costs, (ii) in an adequate timeframe.

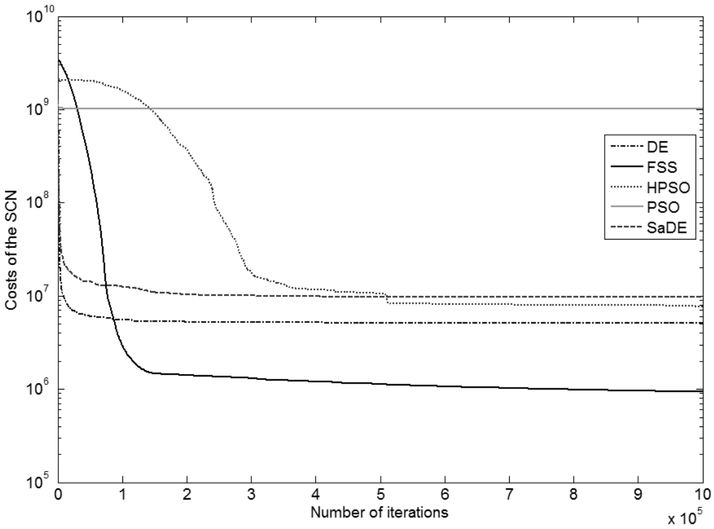

Figure 2: Costs decay for each metaheuristic.

The clearly better results of the algorithm FSS in comparison with the other algorithms regarding the standard deviation shows a higher consistency in its results. From this fact, it can be inferred that the FSS is more likely to avoid local minima and find closer solutions in different runs, and therefore displaying the highest consistency during the solution generation in general. A probable cause is the specialized movement operator of the FSS, which expands the fish school when the search process, as a whole, is not going well, which means that it is stuck in a local minimum. PSO presented, by far, the worst results in this measure. This was expected, since the PSO is well known as an algorithm of very fast convergence, and therefore it is likely to get stuck in local minima. The results also showed that, even though SaDE was the best technique in terms of finding good solutions in a short timeframe, its results are not consistent. It showed a relative tendency of SaDE to be trapped in local minima, in comparison with DE, HPSO and FSS.

Overall, these results show that some of the investigated metaheuristics are applicable to solve the SCNP, creating feasible and cost-efficient plans for the decision problem. The choice of the algorithm allows for a tradeoff between cost effectiveness and the creation time of the plan itself. However, the implemented prototypes show further potential for improvement. When performing a high amount of iterations, the FSS clearly outperforms the other algorithms. The modeling effort for testing different scenarios was found to be similar to that of other optimization approaches, like exact mathematical solvers. Due to the chosen implementation approach, the effort to model new constraints into the planning problem is low compared to other metaheuristic approaches, which exploit the problems characteristics extensively, but thus require adaptations specific to the constraints. Additionally, it has been shown that two out of three of the investigated more advanced methods (HPSO and FSS) outperform the standardized and established ones (PSO and DE). As metaheuristics scale much more favorable with the size of the optimized planning problem than exact optimization techniques, the algorithms inspected in this paper show potential, to be able to solve an enhanced SCNP, which takes further decisions, such as an appropriate choice of the mode of transportation, into account.

6. CONCLUSIONS AND FURTHER RESEARCH

The application of metaheuristics is a promising approach to solve complex SCNP tasks in an applicable timeframe to create mid-term plans, which can be used as a framework for the generation of operational planning tasks such as transport planning. In this work, we have compared five metaheuristics, the established PSO and DE algorithms, as well as the state-of-the-art HPSO, SaDE and FSS algorithms, regarding the quality of the generated solutions and the consistency of the generated results. Overall the state-of-the art approaches have been identified as promising solution methods, as they have generally generated good results. The FSS stands out of this selection, as it shows the highest consistency in terms of the generated master plan's overall cost and feasibility of all algorithms. This however comes at the price of a lower convergence rate, meaning that the time needed to find a good solution, which represents an applicable logistics plan, is higher than that of other techniques. Nevertheless, the inconsistent behavior of the other techniques will also lead to cases, in which a good solution cannot be found in a timely manner.

However, in future work it needs to be determined, if the implementations of the metaheuristics can be improved in order to solve problems of real world size: The solution generation also needs to be fast enough to adhere to the ever shorter planning cycles, that logistic companies face today, and needs to be able to deal with additional requirements and decision options. Furthermore the solution themselves have to be sufficiently close to the optimal solution, in order to create cost-efficient plans, which can then be used as a framework for operational logistics planning tasks such as a more detailed transport planning. To tackle these issues, both, the meta-heuristic and the modeling approach have to be optimized. Additionally, ways have to be found, which lessen the effort needed to include new constraints into the optimization, such as including constraints into the algorithms themselves instead of using modeling approaches such as the inclusion of additional penalty costs. Summarized, the evaluated optimization techniques show the potential to enhance currently used solution techniques, especially regarding the inclusion of further planning decisions, and should be further investigated.

REFERENCES

Bastos-Filho, C. J. A., Lima Neto, F. B., Lins, A. J. C. C., Nascimento, A. I. S., Lima, M. P. (2009) Fish School Search: an overview. *In*: R. Chiong (*ed.*) *Nature-Inspired algorithms for optimisation*. Berlin Heidelberg: Springer. p.261-79.

Bozarth, C.C., Warsing, D., Flynn, B.B. and Flynn, E.J. (2009) The impact of supply chain complexity on manufacturing plant performance. *Journal of Operations Management*, v. 27, n. 1, p.78-93.

Burnwal, S. and Deb, S. (2013) Scheduling optimization of flexible manufacturing system using cuckoo search-based approach. *The International Journal of Advanced Manufacturing Technology,* v. 64, n. 5-8, p. 951-959.

Engelbrecht, A. P. (2010) Heterogeneous Particle Swarm Optimization. *Proceedings of the 7th International Conference on Swarm Intelligence,* ANTS'10, Springer-Verlag, Berlin, Heidelberg, p. 191–202.

Griffis, S.E., Bell, J.E. and Closs, D.J. (2012) Metaheuristics in Logistics and Supply Chain Management. *Journal of Business Logistics*, v. 33, n. 2, p.90-106.

Hellingrath, B. and Küppers, P. (2011) Multi-Agent Based Evaluation of Collaborative Planning Concepts in Heterarchical Supply Chains. *Proceedings of the Logistikmanagement*, p.1-22.

Kennedy, J. and Eberhart, R. (1995) Particle Swarm Optimization. *Proceedings of the IEEE International Conference on Neural Networks*, v. 4, p. 1942-1948.

Marinakis, Y., Iordanidou, G. and Marinaki, M. (2013) Particle Swarm Optimization for the Vehicle Routing Problem with Stochastic Demands. *Applied Soft Computing*, v. 13, n. 4, 1693-1704.

Melián-Batista, B. and Moreno-Perez, J. A. (2013) Introduction to the special issue on intelligent metaheuristics for logistic planning. *Journal of Heuristics*, v. 19, n. 2, p. 97-101.

Ojha, A., Das, B., Mondal, S. K. and Maiti, M. (2013) A multi-item transportation problem with fuzzy tolerance. *Applied Soft Computing*, v. 13, n. 8, p. 3703-3712.

Pibernik, R. and Sucky, E. (2005) Master Planning in Supply Chains. *In*: Günther, H., Mattfeld, D. C. and Suhl, L. (*eds.*) *Supply Chain Management und Logistik*: *Optimierung, Simulation, Decision Support*. Physica-Verlag HD. p.69-93.

Ponsignon, T. and Mönch, L. (2012) Heuristic approaches for master planning in semiconductor manufacturing. *Computers & Operations Research*, v. 39, n. 3, p.479-91.

Qin, A. K. and Suganthan, P. N. (2005) Self-adaptive Differential Evolution Algorithm for Numerical Optimization. *Proceedings of the Congress onEvolutionary Computation*, v. 2, p. 1785-1791.

Rohde, J. and Wagner, M. (2005) Master planning. *In*: Stadtler, H. and Kilger, C (*eds.*) *Supply chain management and advanced planning*. Springer Berlin Heidelberg, p. 159-178.

Simchi-Levi, D., Kaminsky, P. and Simchi-Levy, E. (2007) *Designing and Managing the Supply Chain - Concepts, Strategies and Case Studies*. 3 ed., Mc Graw Hill/Irwing.

Soares, M. M. and Vieria, G. E. (2009) A new multi-objective optimization method for master production scheduling based on genetic algorithm. *The International Journal of Advanced Manufacturing Technology*, v. 41, n. 5-6, p. 549-67.

Stadtler, H. (2005) Supply chain management and advanced planning – basics, overview and challenges. *European journal of operational* research, v. 163, n. 3, p. 575-588.

Storn, R. and Price, K. (1997) Differential Evolution – A simple and efficient adaptive scheme for global optimization over continuous spaces. *Journal of Global Optimization*, v. 11, n. 4, p. 341-359.

Bernd Hellingrath (bernd.hellingrath@wi.uni-muenster.de)
Luis Filipe de A. Pessoa (filipe.pessoa@wi.uni-muenster.de)
Dennis Horstkemper (dennis.horstkemper@wi.uni-muenster.de)
Diego de Siqueira Braga (diego.siqueira@wi.uni-muenster.de)
Chair of Information Systems and Supply Chain Management, Westfälische Wilhelms-Universität Münster
Leonardo-Campus 3, 48149 Münster Germany

Fernando Buarque de Lima Neto (fbln@ecomp.poli.br)
Marcelo Gomes Pereira de Lacerda (mgpl@ecomp.poli.br)
Engenharia da Computação, Escola Politécnica de Pernambuco, Universidade de Pernambuco
R. Benfica, 455 CEP: 50.720-001 – Recife, PE, Brasil